unesp UNIVERSIDADE ESTADUAL PAULISTA

Administração de Redes TCP/IP

Tópico:
Conceitos de Segurança
Parte 1 - Os Ataques e a Reação

Prof. Dr. Adriano Mauro Cansian
adriano@dcce.ibilce.unesp.br
UNESP - IBILCE - São José do Rio Preto

Conceitos de Segurança[1]

Parte 1 - Os Ataques e a Reação

Prof. Dr. Adriano Mauro Cansian
adriano@dcce.ibilce.unesp.br
UNESP - IBILCE - São José do Rio Preto

1. Os Ataques

1.1. Engenharia Social

Podemos definir **engenharia social** como a técnica de obter acesso a computadores utilizando inseguranças de outros meios de comunicação. Estes outros meios podem variar bastante, desde um telefonema para um provedor fingindo ser um usuário legítimo que esqueceu sua senha até uma eventual ida até o balcão bem orquestrada e encenada intimidando ou iludindo um funcionário de uma empresa.

Em um país como o Brasil onde a cultura do "Você sabe com quem está falando?" é tão fortemente enraizada em praticamente todos os níveis da sociedade a engenharia social encontra um mercado potencial.

Ataques comuns e bem sucedidos usando esta técnica tem ocorrido em *chats* e canais de IRC onde um invasor consegue facilmente obter uma conta emprestada de um usuário alegando algum objetivo lícito. Em ambientes acadêmicos é comum o envio de mensagens supostamente de um pesquisador temporariamente afastado da instituição para um administrador inexperiente solicitando a remoção de alguma barreira de acesso ou troca de senha enquanto ele, o pesquisador, encontra-se na outra instituição.

1.2. Roubo de Contas

Terminais de Acesso Público

É muito comum empresas e Universidades disponibilizarem durante feiras comerciais, encontros ou congressos, terminais conectados à Internet para os participantes poderem acessar suas contas remotas. Este tipo de cortesia costuma trazer conseqüências indesejadas

[1] Créditos: este material foi produzido a partir de **"Tópicos em Segurança de Redes"** - Volume I, Versão 0.5.1 , produzido/traduzido por Pedro A. M. Vazquez e outros (27/julho/98). Reproduzido sob permissão de Pedro A.M. Vazquez <*vazquez@iqm.unicamp.br*>.

sob o aspecto de segurança. Invasores de sistemas podem obter acesso físico à rede e roubar senhas utilizando *sniffers*, ou podem simplesmente ficar observando um usuário no momento em que este realiza uma conexão remota e obter a sua senha.

Em salas públicas de terminais em Universidades um invasor pode substituir ou modificar o programa utilizado para *login* por outro que captura a senha digitada por um usuário válido e a guarda para posterior recuperação ou a envia para uma conta remota do atacante

Senhas Fáceis

Algumas instituições, comerciais e acadêmicas, permitem ou fornecem senhas fáceis aos seus usuários visando acelerar e simplificar o acesso destes aos equipamentos. Em alguns casos esse tipo de procedimento de risco está associado a um contrato entre um provedor de acesso e uma escola, por exemplo, o que o leva a abrir um número elevado de contas para os alunos ou, situação semelhante, em instituições de ensino em início de ano letivo logo após as matrículas.

Um invasor bem motivado pode obter uma lista de usuários dessa instituição através de análise de tráfego de e-mail ou mesmo uma simples visita à *home page* desta e utilizar um programa que automatize a procura de senhas por tentativa e erro baseadas no *username* e dados pessoais do usuário (ex. *username* JOAO senha XJOAO).

Senhas Muito Difíceis

Alguns administradores tentam evitar a adivinhação de senhas utilizando métodos que impedem que o usuário utilize senhas fáceis, adicionalmente alguns administradores adotam programas que geram senhas fortes para os usuários mas de difícil memorização. Por último, para evitar problemas com contas eventualmente inativas ou abandonadas alguns administradores também adotam técnicas de expiração periódica de senhas. Em muitos casos alguns usuários administrativos, ou não, necessitam acessar três ou quatro contas sob esta política diariamente para o seu trabalho. O resultado é que terminam anotando estas senhas em pedaços de papel que, por comodidade, são afixados ao monitor ou deixados ao lado do terminal. Basta ao invasor uma visita ao usuário para obter as três ou quatro contas de acesso aos sistemas de contabilidade, controle de notas, etc.

1.3. Vulnerabilidades

As vulnerabilidades nos computadores podem ter quatro origens principais:

- Vulnerabilidades do próprio sistema operacional, seja por projeto ou *bug*;
- Vulnerabilidades introduzidas pela instalação de softwares cuja execução ou operação criam oportunidades para invasores;
- Vulnerabilidades introduzidas pela instalação ou configuração errada do sistema operacional, de softwares ou da rede onde se encontra;
- Vulnerabilidade de acesso físico ao equipamento.

Vulnerabilidades do Sistema Operacional

Apesar da extensa literatura especializada apontando vulnerabilidades históricas em sistemas operacionais, muitos fabricantes persistem, também historicamente, em erros que comprometem totalmente os seus produtos no que diz respeito à segurança.

Desde sistemas que vêm de fábrica com versões arcanas do senmdail até sistemas que ainda possuem um "+" no ***/etc/hosts.equiv*** as vulnerabilidades desta natureza são incontáveis e não param de ser (re)descobertas semanalmente.

Vulnerabilidades dos Softwares

Grande parte dos programas não são projetados tendo a segurança como um dos objetivos. Se estes programas executam com algum privilégio e uma vulnerabilidade é encontrada neles permitindo a sua utilização por um invasor, muitas vezes já é tarde demais para que as correções aplicadas o tornem seguro definitivamente pois a falha está no projeto inicial e não em partes isoladas dele. Os programas ***sendmail*** [8] e ***rdist*** [9] são um exemplos típicos disto.

Além de não visarem a segurança em seu projeto muitos programas são realmente mal escritos e permitem que um invasor localize situações que levem a *stack overflow* desviando a sua execução para rotinas que permitam que ele tenha acesso privilegiado, ou que eles acessem arquivos ou os criem com permissões perigosas.

O sistema UNIX possui mecanismos de segurança bastante efetivos mas poucos programas os exploram . Muitos programas executam com privilégios excessivos e estão disponíveis para todos os usuários como o *SuperProbe* distribuído ***suid root*** junto com o XFree86 até recentemente.

Vulnerabilidades Oriundas da Instalação

Muitos sistemas operacionais possuem sistemas de instalação *user friendly* projetados pelo fabricante para facilitar ao máximo a tarefa do seu cliente. Infelizmente o fabricante não pode prever exatamente o ambiente de trabalho em que o seu sistema será utilizado. Isto reflete-se em *defaults* de instalação extremamente versáteis e liberais. A conseqüência, muitas vezes, é a instalação de um sistema operacional, com níveis de segurança precariamente aceitáveis para operação doméstica, conectando-o diretamente à Internet.

Uma vulnerabilidade de instalação muito comum é aquela produzida pelo marketing do fabricante convencendo o usuário que o seu sistema é extremamente seguro e que não há com que se preocupar. Este tipo de marketing é extremamente convincente em instalações em início de operação (muitas vezes sendo fator determinante de opção de compra) ou naquelas onde a equipe está migrando de um ambiente onde não havia acesso a redes. Este tipo de atitude por parte de um fabricante, em 1996, chegou ao ponto de um grupo de *crackers* criar uma página WWW dedicada ao *bug of the week* desse sistema operacional.

Acesso Físico

Não é raro encontrar-se redes acadêmicas onde todas as medidas de segurança prescritas foram adotas e onde foram aplicados todos os *patches* do fabricante e dos softwares instalados. Não é raro também que o cauteloso administrador dessa rede, por motivos de orçamento, tenha que compartilhar a sua sala de operação com os alunos do departamento e que estes tenham acesso físico livre e franco às unidades de backup, ao console do servidor e ao próprio servidor nos períodos noturnos e finais de semana quando o administrador encontra-se ausente. A mesma situação pode ser encontrada em algumas instalações comerciais e basta um invasor infiltrar-se fisicamente no ambiente para ter plenas possibilidades de acesso garantidas.

Da mesma forma que o marketing do sistema dito seguro põe em risco um servidor o marketing da administração superior a respeito da qualidade do sistema de controle de acesso físico e vigilância pode por em risco uma instalação computacional.

Não é raro observarmos um funcionário legítimo parado na portaria de um CPD apresentando ou entregando todos os seus documentos à vigilância enquanto na porta ao lado entram e saem vendedores, representantes e técnicos muito bem vestidos e sorridentes que se supõem serem de alguma grande empresa do ramo carregados de equipamentos e pastas de documentos.

Muitas vezes as instituições adotam sistemas de crachás magnéticos, portões comandados por leitoras ópticas, restrições de acesso aos funcionários delas mesmas mas deixam abertos os caminhos para um invasor bem vestido e equipado com os quesitos esperados (celular, laptop, crachá de multinacional, etc) passar ao largo. Ou deixam a janela do piso superior aberta. Littman [10] descreve detalhadamente como estes tipos de medidas são ineficazes e como foram exploradas durante uma década por Poulsen para obter acesso físico irrestrito a vários prédios da Pac Bell.

1.4. Exploração de Relações de Confiança

Muitas redes estabelecem relações de confiança entre computadores tentando facilitar o serviço administrativo ou o acesso dos usuários. Estas relações podem ser utilizadas por um invasor para obter acesso ilícito aos computadores concentrando os seus ataques em um computador menos protegido e utilizando-o posteriormente para acessar os computadores de ambas as redes.

Usuários *Dual Homed*

Usuários comuns a duas redes distintas (ex. Universidade A onde trabalha e Universidade B onde realiza projeto de pesquisa em colaboração ou o consultor da empresa X que temporariamente realiza serviços na empresa Y) podem configurar um
arquivo **.rhosts** em uma ou ambas redes que permite acesso transparente sem a necessidade de uso de senhas para autenticação.

Administração via .rhosts

Muitos administradores de redes com um número elevado de computadores optam por estabelecer relações de confiança via .rhosts entre máquinas para simplificar suas tarefas de rotina. O acesso root a uma ou mais máquinas automaticamente garante o acesso root às demais, para eles e para o invasor.

Outros administradores em ambientes de redes UNIX heterogêneas costumam permitir acesso root e suid root via NFS para contornar peculiaridades de operação dos seus sistemas.

Um invasor pode descobrir estas relações de confiança mútua entre os computadores e explora-las de diversas formas. Na eventualidade de uma das redes com relação de confiança estar inacessível (ex. parada de 6 horas para manutenção preventiva) ele pode configurar um computador em uma terceira rede para aparentar-se como um dos computadores da rede inacessível e utiliza-lo para invadir a outra rede.

O invasor também pode investigar as falhas de segurança de uma das máquinas administrativas que possui acesso privilegiado, invadi-la e obter acesso a todas as demais via .rhosts, ou ele pode explorar a configuração de NFS para realizar o mesmo tipo de ataque. Recentemente relações de confiança via .rhosts foram exploradas através da técnica de IP *spoofing* para obter acesso a uma rede do SDSC [12].

1.5. Cavalos de Tróia, *Backdoors*, Vírus, etc

Alguns ISP não fornecem acesso *shell* aos seu usuários por medida de segurança...

..mas deixam os usuários redirecionarem seu e-mail para outros lugares através do mecanismo de ***forward*** de *sendmail*.

Um invasor pode utilizar isso para trocar o seu *shell* para um *shell* válido e obter acesso interativo ao sistema.

No exemplo abaixo o invasor alterou o .forward de uma outra conta para fazer este serviço, e só foi descoberto após a sua conta ter sido encerrada no ISP.

```
  ----- The following addresses had permanent fatal errors -----
chsh onailuig /bin/bash
   (expanded from: <listas@MX.YYY.com.br>)

  ----- Transcript of session follows -----
550 /home/listas/.forward: line 1: chsh onailuig /bin/bash...
User unknown
```

O invasor poderia ter colocado um script no arquivo *.forward* que compilasse *backdoors* ou realizasse tarefas após o recebimento de um e-mail remoto.

Cavalo de Tróia com engenharia social: <ZeRoC00l> ehehehe...

Diálogos deste tipo são bastante comuns no IRC

<yyz> heheh
*** BLaSTeR (Wh0Kn0w?@irc.XXX.com.br) has joined channel #hackers
<ZeRoC00l> ehehehe...
<ZeRoC00l> to c/ programinha legal aki...
<ZeRoC00l> pra quebrar a senha do kaly
<yyz> hehehe...
<ZeRoC00l> kem ker uma copia pvt me...
<yyz> ./me naum quer naum
<BLaSTeR> eu kero!

... e pronto, o **BLaSTeR acaba de transferir um cavalo de Tróia para o seu computador** que é capaz de realizar várias ações, inclusive, talvez, quebrar a senha do kaly.

CLICK HERE TO DOWNLOAD

Basta um click em CLICK HERE TO DOWNLOAD e o arquivo transferido cegamente do WWW pode ser um arquivo PostScript [14] que se visualizado de forma insegura pode comprometer dados vitais do usuário ou criar um arquivo .rhosts que autoriza o seu acesso sem necessitar de senha.

Ou pode ser uma das muitas formas de ataque similar permitidas pelos recursos disponibilizados recentemente no WWW utilizando Java, ActiveX.

Ou pode ser um arquivo executável com a última versão daquele antivírus ou daquele programa de acesso ao IRC que foram devidamente contaminadas a favor de um invasor.

Ou pode ser um documento do Word contendo um vírus de macro, ou do Excell enviado junto com o anúncio daquele congresso alegando ser um formulário pronto para preencher.

1.6. Infraestrutura de Redes, Serviços e Protocolos

1.6.1. Routing tricks

O protocolo de rede usualmente empregado em redes locais é o RIP nas versões 1 e 2. Este protocolo não possui qualquer forma de autenticação. Um atacante pode fabricar um pacote RIP e envia-lo aos roteadores de uma instituição enganando-os completamente.

1.6.2. DNS Spoofing

O DNS tem uma serie de vulnerabilidades [16,17] que podem ser utilizadas por um invasor [18]. Um invasor pode obter acesso ao servidor DNS de uma organização e alterar os mapas deste de forma que o seu computador passe a possuir o nome de um computador desta organização que possui relação de confiança com o computador alvo.

Um invasor pode corromper o servidor DNS de uma organização fazendo que ele responda com o IP de seu servidor WWW quando os usuários da instituição desejavam o IP daquele banco que permite movimentar a conta via rede.

Versões mais antigas de servidores DNS permitem que o seu cache seja corrompido remotamente com os mesmos objetivos e resultados finais

1.6.3. ICMP

Ataques ICMP visam enganar roteadores e gateways enviando informações falsas para estes a respeito da conectividade de computadores e redes, um atacante pode desativar o acesso a um computador enviando pacotes ICMP a outro computador indicando que o primeiro está *unreacheable.*

1.6.4. Source Routing

O protocolo IP especifica que pacotes contendo informação de roteamento devem retornar à sua origem utilizando o caminho reverso de rotas indicado por ele.

Um atacante pode desativar um computador que possui relação de confiança com o computador alvo (utilizando algum ataque como ***Syn flood, nuke,*** etc) e definir em seu computador o IP do computador desativado.

A seguir ele envia pacotes com informação de source routing para o computador alvo. O computador alvo permite o acesso ao invasor baseado na relação de confiança entre ele e o computador desativado.

1.6.5. IP Spoofing

Sob este nome incluem-se várias técnicas de invasão que exploram a autenticação ou acesso e filtragem de pacotes baseadas em endereço IP.

Uma técnica simples de IP spoofing consiste no invasor configurar em um computador o mesmo IP de outro computador que possui uma relação de confiança com o computador alvo do ataque.

O computador cujo IP está sendo falsificado pode estar desligado para manutenção preventiva, pode ter sido desativado pelo invasor utilizando alguma falha do seu sistema operacional, etc. Em alguns casos, como o protocolo UDP, o invasor pode fabricar pacotes falsos utilizando o IP do computador com relação de confiança.

Predição TCP

Este ataque explora uma vulnerabilidade do protocolo TCP. Ele foi previsto inicialmente por Morris [19] em 1985 e analisado por Bellovin [20] em 1989 mas somente em 1994 [11,12] verificou-se a primeira aplicação prática do mesmo.

Este ataque consiste em explorar a predicabilidade do ***Initial Sequence Number*** (ISS) utilizando por um computador durante o *handshake* usado pelo protocolo TCP na seqüência inicial de criação de uma conexão [21].

Sejam os computadores **A** e **B** iniciando tal conexão, neste caso temos:

A -------- SYN:ISSa ----- → B

A ← SYN + ACK(ISSa) : ISSb------ B

A --------ACK(ISSb)----- → B

Embora espere-se que os *ISS* utilizados sejam aproximadamente aleatórios, **algumas implementações TCP/IP utilizam mecanismos bem definidos de geração destes números o que os torna previsíveis, possibilitando um ataque**.

O invasor **X** inicia uma conexão real contra o alvo **B**, por exemplo à porta SMTP. Isto lhe fornece **ISSb**.

A seguir ele envia um pacote falsificado como sendo o computador **A**:

A ----SYN:ISSx--------- → B

O computador B responde à solicitação enviando para A:

A ← ---- SYN + ACK(ISSx):ISSb'-- B

O invasor nunca receberá este pacote mas se ele tiver condições de prever ISSb' poderá falsificar outro pacote como sendo do computador A e envia-lo para B:

A ---- ACK(ISSb')--- → B

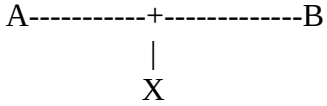
se o invasor previu corretamente **ISSb' ele conseguiu estabelecer uma conexão com B, ele jamais receberá os pacotes enviados por B mas poderá executar comandos em B**, por exemplo, no caso de uma conexão rsh: **echo + + >/.rhosts**.

Além da predição correta do ISS uma outra dificuldade deste ataque é que o computador A, se receber as respostas de B, encerrará o processo enviando um pacote com o flag RST ativo em resposta abortando o ataque.

Isto pode ser contornado explorando vários fatores, desde uma manutenção preventiva prolongada e previamente anunciada no computador A até a sua desativação por parte do atacante utilizando alguma vulnerabilidade do sistema operacional (nuke, ping) ou um ataque *denial of service* como o ***SYN Flood*** descritos a seguir.

1.6.6. TCP Splicing

Um invasor localizado no computador **X** entre duas redes, onde encontram-se os computadores **A** e **B** pode realizar um ataque utilizando esta técnica:



O atacante monitora o usuário em A iniciar uma conexão com B e grava os **ISSa/ISSb** utilizados durante o login.

A seguir o atacante derruba o computador A ou inviabiliza o seu caminho até B.

O usuário em A apenas percebe a perda de conexão com B. O atacante assume o papel de A enviando os pacotes no lugar deste com a numeração correta do ISS.

O sistema B não percebe o ataque.

1.6.7. FTP Bouncing

Um invasor pode localizar um servidor FTP *anonymous* situado atrás de um *firewall* que possui uma área *incoming* aberta para a escrita e utiliza-lo para atacar um alvo ao qual não possui acesso direto mas que pode ser alcançado a partir do servidor FTP *anonymous*:

1. O atacante cria um arquivo contendo um diálogo do protocolo SMTP (por exemplo) e o transfere para a área *incoming*.
2. Em seguida ele realiza um novo ftp para o server e envia o comando PORT contendo o endereço IP do computador alvo e a porta SMTP.
3. O servidor FTP abre uma conexão com o computador alvo e notifica com OK o atacante.
4. O atacante em seguida envia um comando RETR especificando o arquivo contendo o diálogo SMTP previamente transferido para a área incoming.
5. O servidor FTP envia o arquivo para a porta SMTP do computador alvo.

Este tipo de ataque permite que o servidor FTP seja utilizado para enviar um *fake mail* para o alvo sem revelar o endereço de origem do atacante. Dependendo do conteúdo do arquivo enviado pode ser tentado um ataque ao sendmail.

Este ataque pode utilizar outras portas que não a SMTP. Este ataque permite contornar regras de acesso direto impostas por *firewalls* e *packet filters*.

1.6.8. Proxies e Servers Clandestinos

Um atacante com acesso legítimo a um computador em uma rede pode instalar nele um *proxy* como o *WinGate* sem que isso seja percebido pelo administrador.

Como esse computador, em princípio, não aceita conexões externas ele usualmente está fora dos controles e filtros da rede local e usualmente é permitido o acesso dele aos servidores.

O atacante pode utilizar o proxy para, a partir de um computador externo, realizar uma conexão para os servidores internos contornando os controles de acesso existentes.

1.7. Denial of Service

O objetivo destes ataques é **inviabilizar a operação de um site, de um computador ou de um backbone** utilizando algum serviço ou protocolo válido para isto.

Alguns ataques desta natureza podem ser filtrados nos roteadores ou gateways, outros não.

1.7.1. Vulnerabilidades do Sistema Operacional

Nuke

Em 1997 foi descoberto que alguns sistemas operacionais abortam a sua execução e travam ou reinicializam se lhes for enviado um único pacote com os devidos flags TCP ativados.

Big Ping

Em 1997 [22 também foi descoberto que o envio de pacotes ICMP echo com um dado tamanho pode produzir em vários sistemas (incluindo impressoras de rede) os mesmo efeitos do Nuke.

1.7.2. ICMP Flood

O protocolo ICMP é utilizado para controle dinâmico da Internet, routers e hosts fazem uso dele para redirecionar tráfego, avisar que um computador está caído, ou que não há rota para ele, etc.

Este protocolo, como o UDP, permite que se utilize endereços falsos de origem do pacote dificultando a localização da fonte de ataque.

Um atacante pode explorar esse protocolo para inviabilizar a operação de uma rede ou de uma ou várias máquinas enviando para outros roteadores dos quais essa rede ou máquina dependem informações falsas tipo ***host unreacheable***.

O protocolo ICMP também pode ser empregado para produzir floods usando o ping que resultem na saturação de um ou mais links de um backbone.

A Internet/Br sofreu um ataque deste tipo em dezembro de 1996 quando dois floods foram disparados, um a partir da UFRJ e outro a partir da UFPR, contra um provedor localizado no backbone da Embratel.

Mesmo a filtragem realizada pelo POP-RJ da RNP não impediu que o caminho RJ-SP-PR fosse inviabilizado durante a duração do ataque.

Outra técnica, denominada **smurf** [24], explora uma particularidade de alguns roteadores e gateways em transmitir para sua rede interna pacotes ICMP echo destinados ao endereço de broadcast da mesma fazendo com que todos os hosts nela localizados respondam simultaneamente saturando o link.

1.7.3. Mail Bombing

Um atacante pode resolver inviabilizar a operação de um servidor SMTP enviando dezenas (ou centenas) de milhares de mensagens para um desafeto que possui conta nesse servidor.

Ele pode utilizar diversos endereços de origem diferentes dificultando a sua filtragem e ele pode utilizar diversos outros servidores SMTP como relays dificultando ainda mais o bloqueio ao seu ataque.

1.7.4. SYN Flood

O TCP utiliza um protocolo de *handshake* para iniciar novas conexões onde o flag **SYN** é utilizado no pacote inicial para começar, no computador remoto, o estabelecimento da conexão:

local ----SYN---→ remoto
local ←-SYN+ACK- remoto
local ----ACK---→ remoto

Após responder ao **SYN** inicial o computador remoto cria uma conexão embrionária e aguarda um certo intervalo de tempo pela resposta do computador local. Caso ela não venha, essa conexão embrionária é descartada.

O número de conexões embrionárias nos sistemas é limitado**. Um atacante pode enviar um número grande de pacotes com o flag SYN ligado, solicitando uma nova conexão, até exaurir o número de conexões embrionárias suportadas**.

A partir deste instante o sistema remoto não responderá a novos pedidos de conexão. O atacante pode fabricar pacotes com endereço de origem falso e aleatório impedindo a identificação da origem do ataque. O atacante pode manter o envio de pacotes por um período longo de tempo retirando de operação esse computador [25].

1.7.5. Ataques Misteriosos

Estes são os ataques bem sucedidos que ninguém conhece a técnica empregada ou não a identificou ainda. Exemplos de ataques desta natureza são o worm em 1988 [13] e o uso do IP Spoofing em 1994 [12].

Até a determinação da técnica empregada criaram muita especulação a respeito de sua natureza e forma de ação.

2. Referências do tópico 1 - Os Ataques:

1. Littman, J., The Fugitive Game, Little, Brown & Co.,Boston, EUA, 1997.

2. Winkler I.S., Dealy, B., Information Security Technology? Don't Rely on It. A Case Study in Social Engineering, The Fifth USENIX UNIX Security Symposium, Salt Lake City, Utah , 1995.

3. Hafner, K., Markoff, J., Pengo And Project Equalizer, in Cyberpunk, Touchstone, EUA, 1995.

4. Stoll, C., The Cuckoo's Egg, Simon & Schuster Inc., EUA, 1990.

5. Exemplos:
ftp://info.cert.org/pub/cert_advisories:
CA-90:07.VMS.ANALYZE.vulnerability
CA-95:09.Solaris.ps.vul
http://www.auscert.org.au/information/advisories/aus_1997.html:
AA-97.19 SGI IRIX df Buffer Overrun Vulnerability
http://www.zdnet.com/pcweek/news/1216/18ent.html
NT vulnerable to attack on CPU

6. Exemplos:
ftp://info.cert.org/pub/cert_advisories:
CA-93:06.wuarchive.ftpd.vulnerability
CA-94:11.majordomo.vulnerabilities
CA-96.06.cgi_example_code
CA-96.22.bash_vuls CA-97.01.flex_lm
CA-97.07.nph-test-cgi_script
CA-97.09.imap_pop

7. Por exemplo, ftp://info.cert.org/pub/cert_advisories:
CA-92:01.NeXTstep.configuration.vulnerability
CA-94:03.AIX.performance.tools
CA-94:10.IBM.AIX.bsh.vulnerability
CA-96.10.nis+_configuration

8. Por exemplo,
ftp://info.cert.org/pub/cert_advisories/obsolete_advisories/:
CA-90:01.sun.sendmail.vulnerability
CA-93:16.sendmail.vulnerability
CA-93:16a.sendmail.vulnerability.supplement
CA-94:12.sendmail.vulnerabilities
CA-95:05.sendmail.vulnerabilities
CA-95:11.sun.sendmail-oR.vul
ftp://info.cert.org/pub/cert_advisories:
CA-95:08.sendmail.v.5.vulnerability
CA-96.20.sendmail_vul
CA-96.24.sendmail.daemon.mode
CA-96.25.sendmail_groups
CA-97.05.sendmail

9. Por exemplo,
ftp://info.cert.org/pub/cert_advisories/obsolete_advisories/:

CA-91:20.rdist.vulnerability
CA-94:04.SunOS.rdist.vulnerability
ftp://info.cert.org/pub/cert_advisories:
CA-96.14.rdist_vul
CA-97.23.rdist

10. Littman, J., The Watchman, Little, Brown & Co.,Boston, EUA, 1997.

11. CA-95:01.IP.spoofing.attacks.and.hijacked.terminal.connections
ftp://info.cert.org/pub/cert_advisories

12. Shimoura, T., Markoff, J., Takedown, Hyperion, EUA, 1996.

13. Ver por exemplo
ftp://ftp.win.tue.nl/pub/security/worm.report.ps.gz

14. CA-95:10.ghostscript
ftp://info.cert.org/pub/cert_advisories
15. CA-94:07.wuarchive.ftpd.trojan.horse
ftp://info.cert.org/pub/cert_advisories

16. CA-96.04.corrupt_info_from_servers
ftp://info.cert.org/pub/cert_advisories

17. Vixie, P., DNS and BIND Security Issues, The Fifth USENIX UNIX Security Symposium, Salt Lake City, Utah , 1995.

18. Bellovin, S., Using the Domain Name System for System Break-ins, The Fifth USENIX UNIX Security Symposium, Salt Lake City, Utah , 1995.

19. R.T. Morris, Bell Labs Computer Science Technical Report #117, February 25, 1985. ftp://ftp.research.att.com/dist/internet_security/117.ps.Z

20. Bellovin, S., Security Problems in the TCP/IP Protocol Suite, Computer Communications Review 19:2, April 1989.
ftp://ftp.research.att.com/dist/internet_security/ipext.ps.Z

21. Bellovin, S.,
ftp://ftp.research.att.com/dist/internet_security/seqattack.txt

22. Kenney, M., Nasty Ping Problems
http://www.sophist.demon.co.uk/ping/
http://prospect.epresence.com/ping

23. CA-96.01.UDP_service_denial
ftp://info.cert.org/pub/cert_advisories

24. Huegen, C. A., The Latest in Denial of Service Attacks: smurfing Description and Information to Minimize Effects
http://www.quadrunner.com/c-huegen/smurf.txt

25. CA-96.21.tcp_syn_flooding
ftp://info.cert.org/pub/cert_advisories

3. A Reação

3.1. Introdução

Este texto [1] faz algumas sugestões para tornar a sua máquinaUNIX mais segura após ela ter sofrido uma invasão. Mesmo que suas máquinas não tenham sido comprometidas ele contém várias dicas úteis para torna-las mais seguras.

3.2. De onde Veio o Ataque

→ Tente traçar ou descobrir a origem do invasor analisando:

who	w
last	lastcomm
netstat	snmpnetstat
syslog	wrappers
logs do roteador	maillog
logs dos wrappers	finger
.history	.rchist

e similares, além dos diretórios /var/logs , /var/adm e /usr/logs

Observação: who, w, last, e lastcomm são comandos que dependem dos arquivos /var/adm/pacct, /usr/adm/wtmp, e /etc/utmp para prover as informações para você. A maioria dos ***backdoors*** deixará o invasor de fora desses arquivos.

Mesmo que o invasor não tenha instalado nenhum *backdoor* ainda, é trivial remover qualquer detecção desses logs.

Mas pode acontecer do invasor esquecer de um ou dois deles. Especialmente se você possui arquivos adicionais e não padronizados no seu sistema.

Sugestão: Instale o xinetd ou TCP Wrapper, que irão logar todas as conexões para a sua máquina permitindo que você descubra se há alguém tentando acessa-la.

Envie a saída do syslog para outra máquina dificultando que o invasor detecte os logs e os modifique.

Outra possibilidade: **netlog** [3]

Pode ser mais interessante monitorar o invasor via algum tipo de *sniffer ethernet* para descobrir que vulnerabilidade(s) ele está explorando nos seus sistemas antes de tomar medidas corretivas.

→ Feche o acesso externo à máquina comprometida.

Remova-a da rede para bloquear completamente o acesso do invasor.

Se o invasor descobrir que o administrador está atras dele, ele pode tentar apagar suas pegadas com **rm -rf /**

3.3. Verificando a Integridade do Sistema

Verifique a integridade dos executáveis em relação a cópias originais. Em particular verifique os seguintes programas porque eles são os mais comumente substituídos por versões contendo *backdoors* para garantir o acesso posteriormente:

1. **/bin/login**
2. todos os servers de rede e tudo que for chamado via inetd
3. **libc.so.***

Outros programas que costumam ser substituídos por versões com backdoor:

1. **netstat** para ocultar conexões
2. **ps** para ocultar processos (Crack, sniffers)
3. **ls** para ocultar arquivos e diretórios
4. **ifconfig** para ocultar interfaces em modo promíscuo
5. **sum** para enganar o resultado do *checksum* de binários que foram substituídos.

Utilize **ls -lac** para **obter a data real de modificação dos arquivos**. Verifique */etc/wtmp* (se você ainda tiver um) com relação a qualquer alterações no relógio do sistema.

Verifique os arquivos em relação aos originais da distribuição (CD ou fita) ou calcule os *checksums* MD5 e compare com os originais que você mantém *off line* (Você os calculou algum tempo atrás, não?).

Sugestão: rode o **cmp** em relação à cópias que você tem certeza a respeito da integridade.

Outro *backdoor* muito comum consiste em transformar um comando comum em suid root para obter acesso root a partir de contas regulares. Para descobrir todos os arquivos suid use:

```
find / -type f -perm -4000 -ls
```

Você pode chegar a conclusão que será necessário reinstalar todo o sistema operacional para ter certeza que não contém nenhum *backdoor*.

Neste caso rode o **Tripwire** após a reinstalação, ele previne que arquivos do sistema sejam modificados sem o conhecimento do administrador

3.4. Senhas

Implemente algum mecanismo de senhas para os seus usuários que verifique que eles as trocam com frequência.

Instale o anlpasswd, npasswd, ou passwd+ no lugar do passwd (ou yppasswd) de forma que eles sejam forçados a escolher senhas razoavelmente seguras.

Utilize o Crack [4] para ter certeza que os seus usuários não estão passando por cima da verificação de senhas fracas. O Crack assegura que os usuários estão escolhendo senhas difíceis. Estando em uma rede o tráfego de senhas na forma de texto plano é um problema.

Algumas opções para contorna-los são: switches Ethernet (impedem o sniffing), tecnologias OTP (One Time Password) como o S/Key [6] ou de criptografia como o ssh [5].

3.5. As Contas

Verifique os arquivos .rhosts e .forward de todos os seus usuários e certifique-se que nenhum deles possui conteúdo suspeito ou fora do comum.

Se o arquivo .rhosts contiver '+ +', a conta pode se acessada de qualquer lugar da Internet.

O **COPS** possui scripts que verificam o .rhosts.

```
find / -name .rhosts -ls -o -name .forward -ls
```

Procure e liste, também, todos os arquivos criados ou modificados ao longo do intervalo de tempo que você suspeita que a invasão ocorreu:

```
find / -ctime -2 -ctime +1 -ls
```

Encontrará todos os arquivos modificados a não menos que um dia mas não mais que dois dias atrás.

Também vale a pena dar uma verificada em todos os arquivos .**login, .logout, .profile, .cshrc** (ao menos a data/hora de modificação).

Certifique-se que não existem arquivos **.rhosts** para contas especiais ou bloqueadas como **news, sundiag, sync**, etc.

O *shell* dessas contas deve ser algo como **/bin/false**, de qualquer forma, (e não */bin/sh*) para torna-las mais seguras.

Também procure por diretórios com nomes do tipo **“. ” (ponto-espaço), “.. ” (ponto-ponto-espaço), “. .” (ponto-espaço-ponto)** e similares.

Eles usualmente são criados no **/tmp**, **/var/tmp**, **/usr/spool/*** e em qualquer lugar que possua permissão de escrita pública.

3.6. Filesystems

Verifique se os *filesystems* que você exporta via NFS não são de escrita pública por qualquer um.

O **NFSWatch** de David Curry [7] loga todas as transações NFS que estão ocorrendo na sua rede.

Verifique com o comando **showmount -e** que o seu sistema está exportando realmente o que você espera e para onde.

Existem *bugs* em algumas implementações do nfsd que ignoram as listas de acesso quando elas excedem algum limite (256 bytes).

Verifique também o que você está IMPORTANDO!!! Use o **flag nosuid** sempre que possível.

Você não deseja ser invadido por um administrador de outro computador (ou um *cracker* usando o outro computador) que possa executar programas suid montados via NFS, deseja?

3.7. *Patches* e Correções

Certifique-se que você está rodando a última versão do *sendmail*. Versões antigas permitiam a execução remota de comandos em qualquer máquina UNIX.

Tente instalar todos os *patches* de segurança do seu vendedor nas suas máquinas.

3.8. Os Alvos Mais Visados

Aqui segue uma lista das vulnerabilidades mais comuns de um computador UNIX:

- Execute um **rpcinfo -p** para certificar-se que a sua máquina não está rodando nenhum programa desnecessário (ex. rexd);

- Verifique que não há um **+** no **/etc/hosts.equiv**;

- Verifique que o **tftpd** está desabilitado.

Se não estiver desabilite ou, ao menos use o **flag -s** para que ele faça um **chroot** para um local seguro se você não puder viver sem ele. Ele é utilizado para boot de Xterms, mas algumas vezes isso pode ser evitado montando via NFS os discos apropriados.

Sob nenhuma circunstância você deve executa-lo como root.

Modifique a linha que o descreve no **/etc/inetd.conf** para algo do tipo:

```
tftp dgram udp wait nobody /usr/etc/in.tftpd in.tftpd -s /tftpboot
```

ou melhor ainda, use o **tcpd wrapper** para proteger o tftpd e logar todas as conexões feitas a ele.

```
tftp dgram udp wait nobody /usr/etc/tcpd in.tftpd -s /tftpboot
```

e edite o ***/etc/hosts.allow*** para restringir o acesso apenas aos clientes dele.

- Verifique **cron** e **at** se não possuem instaladas bombas-relógio que irão explodir depois que você achar que havia removido todas as armadilhas criadas pelo invasor.

- Verifique os arquivos **/etc/rc.boot**, **/etc/rc.local** (no System V olhe **/etc/rc?.d/***) e outros arquivos utilizados na inicialização do sistema.

Confira todos os outros arquivos contendo informações da configuração do sistema (**sendmail.cf, hosts.allow, at.allow, at.deny, cron.allow, cron.deny, hosts, hosts.lpd**, etc). No arquivo aliases procure por aliases que expandem para programas (ex. uudecode).

- Verifique se o inetd.conf e o **/etc/services** não contém nenhum serviço novo adicionado pelo invasor.

- Copie todos os arquivos de log que você possui (**pacct, wtmp, lastlog, sulog, syslog, authlog** e quaisquer logs adicionais que você gera) para algum local seguro (*off line*) de forma que você possa examina-los posteriormente.

Caso contrário não se surpreenda se eles sumirem no dia seguinte quando o invasor se der conta que esqueceu de remover algum deles. Use a sua imaginação para procurar outros traços que o invasor possa ter deixado para trás. Que tal o **/tmp/***?

- Faça um backup do **/etc/passwd** e guarde *off line* e depois troque todas as senhas de root de todas as suas máquinas (depois de certificar-se que os comandos **su** e **passwd** não foram modificados pelo invasor).

Pode parecer terrível (especialmente se você possui 2000 usuários) mas bloqueie todas as contas colocando um * no campo de password.

Se o invasor possuir uma cópia do arquivo de senhas ele possivelmente descobrirá algumas senhas mais cedo ou mais tarde (é só uma questão de ter um bom dicionário). Na verdade ele pode inclusive ter inserido algumas senhas que somente ele sabe para algum usuário inativo.

Nos servidores NIS verifique não apenas os arquivos **/etc/passwd** e **/etc/groups** reais **mas também aqueles utilizados para gerar os mapas de NIS**.

- Certifique-se que o seu servidor de *ftp anonymous* está corretamente configurado.

- Para facilitar a sua vida da próxima vez (ou se você ainda não conseguiu livrar-se do invasor) pense na possibilidade de instalar o **identd**. Ele pode ser usado em uma rede, junto com o tcpd, para descobrir que contas o invasor está utilizando.

- Certifique-se que o único terminal secure é o console do equipamento.

Isso pode evitar logins root a partir da rede. Pode não ser uma grande medida de segurança pois se alguém sabe a senha do root ele possivelmente também sabe as senhas de outros usuários, mas é melhor do que nada, não?

- Confira que **o hosts.equiv**, **.rhosts, e hosts.lpd** não possuem # como comentários. Se um invasor modificar o hostname da máquina dele para # , ela será considerada uma *trusted host* e o acesso permitido às suas máquinas.

- **Mantenha os relógios (*clocks*) das máquinas ajustados** com alguma base de tempo confiável (sugestão: use ntp). Caso contrário os logs não serão úteis para serem cruzados com *logs* de outras máquinas.

- E lembre-se... Existem tantas maneiras pelas quais alguém pode ter modificado o seu computador que você terá que manter olhos e ouvidos bem abertos por bom tempo. Nas listas acima foram citadas apenas as coisas mais óbvias a serem conferidas.

3.9. Disparando o Alarme

Envie mensagens para todos os sites que você conseguiu descobrir que o invasor utilizou e avise-os. **Entretanto, não comente o incidente com pessoas não autorizadas**. Se você tem um plano de emergência, siga-o (por exemplo, notificar seu superior, desligar a rede, etc...).

Ainda, mande uma cópia para cert@cert.org.

Confira também todos os sites na sua vizinhança, ie. aqueles no seu domínio, instituição, etc. Normalmente é trivial para um invasor acessar outro sistema usando rlogin se este possui um subconjunto dos usuários do primeiro (e utilizando **.rhosts** para facilitar o acesso).

3.10. Firewalls

Uma maneira preventiva de evitar que invasores cheguem a tentar o acesso é através da instalação de um *firewall*.

Efeitos Colaterais: *Firewalls* podem ser caros, a filtragem de pacotes pode reduzir a banda da sua rede. Pense na possibilidade de filtrar nfs (porta 2049/udp) e o portmap (11/udp) no seu roteador. Estes protocolos possuem controle de acesso e autenticação mínimos.

Sugestão: bloqueie todas as portas UDP exceto DNS e NTP. Proíba pacotes com *source routing*. Desligue o *ip-forwarding*.

4. Referências do tópico 3 - A Reação:

1. Compromise FAQ, é produzido por Christopher Klaus (cklaus@iss.net) e ©de Internet Security Systems, Inc. Suite 660, 41 Perimeter Center East, Tel: (770) 395-0150, Atlanta, Georgia 30346, Fax: (770) 395-1972. A versão original mais recente pode ser obtida em: http://www.iss.net/ ou ftp://ftp.iss.net/pub/

2. O texto original possui a seguinte licença

Permission is hereby granted to give away free copies electronically. You may distribute, transfer, or spread this paper electronically. You may not pretend that you wrote it. This copyright notice must be maintained in any copy made. If you wish to reprint the whole or any part of this paper in any other medium excluding electronic medium, please ask the author for permission.

3. netlog, ftp://net.tamu.edu:/pub/security.

4. Crack, ftp://ftp.cert.org:/pub/tools/crack.

5. ssh, http://www.cs.hut.fi/ssh/.

6. O S/Key é marca registrada da Bell Communications Research, Inc.
ftp://ftp.bellcore.com/pub/nmh/

7. NetWatch, ftp://harbor.ecn.purdue.edu:/pub/davy

8. Steps for Recovering from a UNIX Root Compromise
ftp://info.cert.org/pub/tech_tips/root_compromise

9. Intruder Detection Checklist
ftp://info.cert.org/pub/tech_tips/intruder_detection_checklist

10. Intrusion Response Work http://www.EnGarde.com/ mcn/response/

Anotações:

Arquivo: **net-sec1.pdf**
Última atualização: quinta-feira, 14 de novembro de 2002 19:10:51